Journal of Fluids and Structures (2000) **14,** 281–301
doi:10.1006/jfls.1999.0268, available online at http://www.idealibrary.com on IDEAL®

COMPUTATION OF TRANSIENT NONLINEAR SHIP WAVES USING AN ADAPTIVE ALGORITHM

M. S. Çelebi

Faculty of Naval Architecture and Ocean Engineering, Istanbul Technical University, Istanbul, Turkey

(Received 17 July 1998, and in final form 4 October, 1999)

An indirect boundary integral method is used to solve transient nonlinear ship wave problems. A resulting mixed boundary value problem is solved at each time-step using a mixed Eulerian–Lagrangian time integration technique. Two dynamic node allocation techniques, which basically distribute nodes on an ever changing body surface, are presented. Both two-sided hyperbolic tangent and variational grid generation algorithms are developed and compared on station curves. A ship hull form is generated in parametric space using a B-spline surface representation. Two-sided hyperbolic tangent and variational adaptive curve grid-generation methods are then applied on the hull station curves to generate effective node placement. The numerical algorithm, in the first method, used two stretching parameters. In the second method, a conservative form of the parametric variational Euler–Lagrange equations is used the perform an adaptive gridding on each station. The resulting unsymmetrical influence coefficient matrix is solved using both a restarted version of GMRES based on the modified Gram–Schmidt procedure and a line Jacobi method based on LU decomposition. The convergence rates of both matrix iteration techniques are improved with specially devised preconditioners. Numerical examples of node placements on typical hull cross-sections using both techniques are discussed and fully nonlinear ship wave patterns and wave resistance computations are presented.

1. INTRODUCTION

The aim of the present paper is to examine the transient ship-wave problem by taking into account an adaptive algorithm on a continuously changing hull surface, on which the kinematic hull boundary condition must be applied. It is known that transient ship-wave problems in their exact form are quite difficult to solve, due to both nonlinearity of the free-surface boundary conditions and the instantly changing hull surface. A mixed Eulerian–Lagrangian scheme (Longuet-Higgins & Cokelet 1976) is employed to solve the fully nonlinear problem. The basis of the method is to use the free-surface boundary conditions to time-step the solution. At each time-step, the positions of the free surface and the body surface are known. The value of the potential on the free surface (Dirichlet condition) and its normal derivative on the body surface (Neumann condition) are also known. This well-defined mixed boundary value problem can be solved using a variety of numerical methods. The kinematic and dynamic free-surface boundary conditions are then repeated for the next time-step. The indirect desingularized boundary integral method (Cao, Schultz & Beck 1991) is used to solve the mixed boundary value problem. The main advantage of this technique is that singularities are placed outside the fluid domain. This avoids the need to evaluate singular integrals on the body and free surface and allows one to use a simple numerical quadrature. The total computational effort at each time-step is thus greatly reduced. Beck, Cao & Lee (1993) and Beck, Cao, Scorpio & Schultz (1994) used

0889–9746/00/040281 + 21 $35.00/0

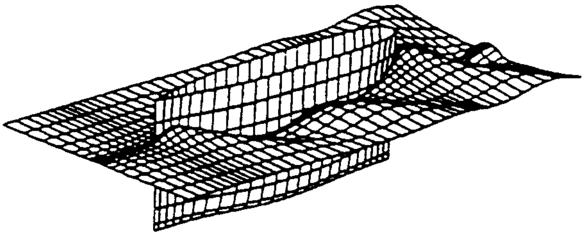

Figure 1. Earlier Wigley hull computations.

a Wigley hull as a typical body form due to its simple mathematical representation. This allowed efficient computation of the hull offsets and unit normal at any point. In addition, the bow and stern profiles were vertical so that computational stations could be fixed along the length. Since the vessel length did not vary as the hull waterline changed, there was no need to change the computational station spacing. For the previous Wigley hull computations, a cosine node spacing in the longitudinal direction and a constant spacing in the vertical direction were used. There was an equal number of nodes at each station (see Figure 1 for an illustration of such a computation).

For arbitrary ship hulls, several differences from the simplified Wigley hull form calculations must be taken into account. Celebi & Beck (1997) studied the nonlinear ship-wave resistance problem for an angled bow and stern of a Wigley hull using an adaptive geometric modelling. Here we will extend this study by adding a new adaptive gridding technique developed by Vinokur (1980), and iterative computation algorithms for the block matrix equations (Celebi *et al.* 1997, 1998). A second boundary value problem for $\delta\phi/\delta t$ is also set up for accurate estimation of fluid particle accelerations surrounding the body. First, in order to account for the arbitrary hull shape, we will use a parametric B-spline hull, surface representation of the hull, and then use a special transformation to move from parametric space to physical space. Second, the girth from the keel line to the waterline can vary significantly in the longitudinal direction. Thus, it is no longer desirable to have the same number of nodes at each station. Third, there is often bow and stern rake, so that the waterline length varies as the hull moves into and out of the water. In order to maintain constant relative spacing between the computational stations, their longitudinal positions along the changing hull length must also vary with time. Finally, the node distribution should be sensitive to station form (i.e., for a given number of nodes, the nodes should be concentrated in regions of high curvature).

2. MATHEMATICAL FORMULATION

2.1. The Mixed Initial Boundary Value Problem

The fluid is assumed inviscid and the flow irrotational, so that a velocity potential (solenoidal vector field) exists in the fluid domain. Surface tension is neglected on the free surface. An Oxyz Cartesian coordinate system is chosen such that the $z = 0$ plane corresponds to the calm water level and z is positive upwards. The coordinate system Oxyz, for the forward speed problem, is translating in the negative x direction relative to a space fixed frame; for the zero forward speed problem it is fixed, and the x–z plane is coincident with the centre-plane of the body. The total velocity potential can then be expressed as

$$\Phi = U_0(t)x + \phi(x, y, z, t), \tag{1}$$

where $U_0(t)$ is the fluid velocity and $\phi(x, y, z, t)$ is the perturbation potential. The boundary value problem is governed by the Laplace equation, and both Φ and ϕ satisfy the Laplace equation. The fluid domain is bounded on top by the free surface (S_F), internally by the hull of the body (S_H), below by a bottom surface (S_B) which is flat, and an enclosing contour surface (S_∞) which includes upstream, downstream and side boundaries. A kinematic body boundary condition is then applied on the wetted surface of the body,

$$\frac{\partial \phi}{\partial n} = -\, U_0(t) n_x + V_H\, \mathbf{n} \quad \text{on } \mathrm{S}_H, \tag{2}$$

where V_H is the velocity of the body including rotational effects relative to the Oxyz coordinate system. A similar kinematic condition must be satisfied on the flat bottom,

$$\frac{\partial \phi}{\partial n} = 0 \quad \text{on } S_B. \tag{3}$$

Both the kinematic and dynamic boundary conditions must be satisfied on the instantaneous free surface. The kinematic free-surface boundary condition requires that the normal velocity of the fluid and of the free surface are equal:

$$\frac{\partial \eta}{\partial t} = -\, \nabla\phi \cdot \nabla\eta + \frac{\partial \phi}{\partial z} - U_0(t)\frac{\partial \eta}{\partial x} \quad \text{on } S_F, \tag{4}$$

where $z = \eta(x, y, t)$ is the free-surface elevation. The dynamic condition requires that the normal stress on the free surface equals the ambient pressure P_a. The dynamic boundary condition, from Bernoulli's equation, then becomes

$$\frac{\partial \phi}{\partial t} = -\, g\eta - \frac{1}{2}|\nabla\phi|^2 - U_0(t)\,\frac{\partial \phi}{\partial x} - \frac{P_a}{\rho} \quad \text{on } S_F, \tag{5}$$

where ρ is the fluid density and g is the gravitational acceleration. The boundary condition applied on the far-field boundary requires that the disturbance vanish at infinity, such that

$$\nabla\phi \to 0 \quad (\text{on } S_\infty). \tag{6}$$

Initial conditions which correspond to starting the body from rest are specified such that the velocity potential on the free surface and the position of the free surface are set equal to those on the calm free surface,

$$\begin{aligned} \phi &= 0 \quad (\text{in fluid domain, at } t = 0), \\ \eta &= 0 \quad (\text{on } S_F, \text{ at } t = 0). \end{aligned} \tag{7}$$

2.2. Time Marching Of The Free Surface Boundary Conditions

The most common approach to time marching the free-surface boundary conditions is a material-node approach in which the nodes or collocation points follow the individual fluid particles. An alternative technique is to prescribe the horizontal-movement-of-the-node but allow the node to follow the vertical displacement of the free surface. The prescribed movement of the nodes may be zero, such that the node locations remain fixed in the x-y plane. Depending on the problem, one of these techniques may be easier to apply than the other. It is convenient to rearrange the kinematic and dynamic boundary conditions in terms of the time derivative of a point moving with a prescribed velocity **v** relative to the Oxyz coordinate system. Both free-surface boundary conditions can then be put in the form

$$\frac{\delta\eta}{\delta t} = -(\nabla\phi - \mathbf{v})\cdot\nabla\eta + \frac{\partial\phi}{\partial z} - U_0(t)\frac{\partial\eta}{\partial x} \quad \text{on } S_F \tag{8}$$

and

$$\frac{\delta\phi}{\delta t} = -g\eta - \frac{1}{2}|\nabla\phi|^2 + \mathbf{v}\cdot\nabla\phi - U_0(t)\frac{\partial\phi}{\partial x} - \frac{P_a}{\rho} \quad \text{on } S_F, \tag{9}$$

where

$$\frac{\delta}{\delta t} \equiv \frac{\partial}{\partial t} + \mathbf{v}\cdot\nabla \tag{10}$$

is the time derivative following the moving node. If $\mathbf{v}$ is set equal to $(U(t), V(t), \delta\eta/\delta t)$ the node follows a prescribed path with velocity $(U(t), V(t))$ in the x–y plane and moves vertically with the free surface. The above equations then reduce to

$$\frac{\delta\eta}{\delta t} = -\nabla\phi\cdot\nabla\eta + U(t)\frac{\partial\eta}{\partial x} + V(t)\frac{\partial\eta}{\partial y} + \frac{\partial\phi}{\partial z} - U_0(t)\frac{\partial\eta}{\partial x} \quad \text{on } S_F \tag{11}$$

and

$$\frac{\delta\phi}{\delta t} = -g\eta - \frac{1}{2}|\nabla\phi|^2 + U(t)\frac{\partial\phi}{\partial x} + V(t)\frac{\partial\phi}{\partial y} + \frac{\delta\eta}{\delta t}\cdot\frac{\partial\phi}{\partial z} - U_0(t)\frac{\partial\phi}{\partial x} - \frac{P_a}{\rho} \quad \text{on } S_F. \tag{12}$$

Setting the fluid velocity $U_0(t) = 0$, we obtain

$$\frac{\delta\eta}{\delta t} = -\nabla\phi\cdot\nabla\eta + U(t)\frac{\partial\eta}{\partial x} + V(t)\frac{\partial\eta}{\partial y} + \frac{\partial\phi}{\partial z} \quad \text{on } S_F \tag{13}$$

and

$$\frac{\delta\phi}{\delta t} = -g\eta - \frac{1}{2}|\nabla\phi|^2 + U(t)\frac{\partial\phi}{\partial x} + V(t)\frac{\partial\phi}{\partial y} + \frac{\delta\eta}{\delta t}\cdot\frac{\partial\phi}{\partial z} - \frac{P_a}{\rho} \quad \text{on } S_F. \tag{14}$$

The last form of the kinematic and dynamic boundary conditions allows the value of the free-surface elevation and potential to be stepped forward in time. One difficulty is the evaluation of the gradient of the free-surface elevation, $\nabla\eta$. Therefore it must be evaluated numerically, but this increases the computer CPU time and sometimes leads to numerical inaccuracies.

The spatial derivatives of η are obtained by using both a piece wise cubic spline function and a finite difference scheme. Here, all node points in the x direction are fitted to a piecewise cubic spline. Then any point other than the node points is found by employing the cubic spline function with the known coefficients. Once we know the new locations of the nodes, we can use the second-order finite difference scheme to obtain the spatial derivative of η. In problems with forward speed, the material node approach has difficulties near the body because nodes tend to pile up near the body stagnation regions, and with material nodes one must always be concerned that the nodes do not penetrate the body surface between time-steps since they are unconstrained. In this study, we use the prescribed horizontal movement-of-the-node approach, which is appropriate for the forward speed problem.

3. ADAPTIVE GRID APPROACH

Two different adaptive curve grid generation techniques are investigated for the node distribution on the ship hull. First, a variational grid generation technique, and second an algebraic grid generation technique, based on the so-called Vinokur (two-sided hyperbolic tangent) distribution.

In a successful adaptive grid, the grid points on the station curves must efficiently distribute themselves so that a functional relationship between these points can represent the physical solution with sufficient accuracy. Data manipulation, communication between nodes, error evaluation and redistribution of nodes in the light of the error evaluation are the basic techniques involved. An adaptive grid approach must contain the following characteristics. (i) The nodes must concentrate in areas of high curvature and yet no region can be allowed to become devoid of nodes. (ii) The distribution must also retain a sufficient degree of smoothness, and the grid must not become too skewed, otherwise the truncation error is increased. This means nodes must not move independently, but rather each node must somehow be coupled at least to its neighbors. (iii) The node points must not move too far, or else unwanted oscillations in the computed results may occur. (iv) Finally, the solution errors must be sensed, and there must be a mechanism for translating this into modification of the grid. The need for a mutual influence among the nodes requires some sort of attraction (repulsion) between node points.

Variational principles provide a clear and intuitive means of building grid generation algorithms and they allow the control of the length of segments in the grid, areas of cells in the grid, and the orthogonality of the angles between grid lines (for surface grids). Weighted variational principles, leading to weighted grid generators, are often more useful due to the use of weight functions for controlling the grid spacing. Brackbill & Saltzman (1982) have first proposed the variational grid generation method for planar 2-D regions. Saltzman (1986) extended the method to an arbitrary surface grid generation later. These methods were formulated in terms of a continuous mapping from the physical object to the logical domain. The motivation for reformulation was direct control over grid qualities such as smoothness, area, and orthogonality. Although largely successful in the plane, the latter approach results in solution bifurcation of the discretized equations when applied to curve and surface grid generation (Steinberg & Roache 1991). Highly curved physical objects caused solution bifurcations that could not be avoided by increased grid resolution or by alternative discretizations of the grid generation equations. The problem of grid generation on lines and surfaces of high curvature appears to have been overcome by reformulating the continuum functional in terms of the physical variables, with constrained minimizations. Castillo (1987) developed a theory of variational grid generation in the plane, known as the direct variational method, in which discrete functionals are directly minimized using nonlinear optimization techniques. In this study, a weighted adaptive grid generator which controls the grid spacing using a space weight function is used.

A two-sided hyperbolic tangent distribution was first proposed by Vinokur (1980). The main advantage of this technique is that it allows the grid distribution to be controlled by two parameters on both sides of the curve. In this way, the node distribution along the station curves can be optimized by two stretching parameters located at both keel and deck ends. This technique is extensively used by some CFD applications and multiple block grid generation algorithms (Steinbrenner *et al.* 1990).

4. VARIATIONAL GRID GENERATION ON STATION CURVES

A variational approach to curve generation is used to create grids with specified distances between the node points. A significant difference between space curve (which is a three-dimensional curved line) and straight line grid generation is that the Jacobian matrix is not square. As a result, constraints must be imposed on the curve minimization to ensure that the resulting grid points lie upon the given station curves.

4.1. Grid Generator That Controls Spacing

In this section, a second-order differential equation for generating grids is introduced, such that the length of the segments in the grid are specified by a *weight function*. The details of this derivation can be seen in Knupp & Steinberg (1994). Consider a weight function $\phi(\xi)$ that depends on the logical-space variable ξ, for ξ in the interval [0, 1]. The lengths of the grid intervals are to be *positive* and *proportional* to ϕ, so it is natural to assume that $\phi = \phi(\xi) > 0$. If M is a positive integer, then an unfolded grid on the interval $[a, b]$ containing $M + 1$ points is given by $x_i, 0 \leq i \leq M$, where $x_0 = a, x_M = b$, and $x_i < x_{i+1}$, $0 \leq i \leq M - 1$. The problem is to generate a grid so that the lengths of the intervals $[x_i, x_{i+1}]$ are proportional to the value of ϕ at the midpoint of the interval. In other words, find x_i so that

$$x_{i+1} - x_i = \kappa\phi\left\{\frac{(\xi_{i+1} + \xi_i)}{2}\right\}, \tag{15}$$

where $0 \leq i \leq M - 1$ and κ is some constant that is to be found. If the grid is given by a transformation from logical to physical space, then $x_i = x(\xi_i)$ where $\xi_i = i/M = i\Delta\xi$. If ϕ is continuous, then for $\Delta\xi$ going to zero, the left-hand side of equation (15) goes to zero while the right-hand side does not. To resolve this, we set $\kappa = C\Delta\xi$ where C is another constant, and divide equation (15) by $\Delta\xi$ to obtain

$$\frac{x_{i+1} - x_i}{\Delta\xi} = C\phi\left\{\frac{(\xi_{i+1} + \xi_i)}{2}\right\}. \tag{16}$$

If x is continuous, then the limit of equation (16) as $\Delta\xi \to 0$ yields the ordinary differential equation

$$x_\xi(\xi) = C\phi(\xi). \tag{17}$$

Dividing equation (17) by ϕ and differentiating with respect to ξ gives

$$\left\{\frac{x_\xi}{\phi}\right\}_\xi = 0. \tag{18}$$

If ϕ is differentiable and x is twice differentiable, the quotient rule for derivatives and some algebra gives

$$x_{\xi\xi} - \left\{\frac{\phi_\xi}{\phi}\right\} x_\xi = 0. \tag{19}$$

The transformation must satisfy the boundary conditions $x(0) = a$, $x(1) = b$. These boundary conditions along with the linear differential equation given in equation (19) uniquely determine the transformation x.

4.2. Parametric Approach

One way to impose constraints which ensure the resulting grid points lie upon the given station curves is to transform the curve minimization into a *parametric space.*

Suppose that an initial parameterization $x(r)$, $0 \leq r \leq 1$, of a curve with continuous second derivatives is given and the curve is to be reparameterized using a variational grid generator. The curve grid functional has the property that the local tangent is proportional to the physical-weight function. We shall minimize the curve functional

$$\mathfrak{I}[\mathbf{x}] = \int_0^1 \frac{\mathscr{H}(g_{11})}{w^2(\mathbf{x})}\,\mathrm{d}\xi, \tag{20}$$

where $\mathscr{H}: \mathbb{R} \rightarrow \mathbb{R}$ is an arbitrary positive real homogeneous function, $g_{11} = \mathbf{x}_\xi \cdot \mathbf{x}_\xi$ is the length metric ($\mathbf{x}_\xi$ represents the derivative of $\mathbf{x}$ with respect to ξ), and w is a physical-space weight function. The set of admissible functions must satisfy the end-point constraints $\mathbf{x}(0) = \mathbf{a} \in \mathbb{R}^3$ and $\mathbf{x}(1) = \mathbf{b} \in \mathbb{R}^3$. A space curve requires three functions $x(\xi)$, $y(\xi)$ and $z(\xi)$ to be found. As a result, a constraint must be imposed on the minimization to ensure that the new parameterization, $\mathbf{x}(\xi)$, is a reparameterization of the original curve $\mathbf{x}(r)$; i.e., the new points must lie upon the curve implied by the old parameterization.

One method of implementing the constraint is the Lagrange multiplier approach and the other is called the parametric approach. We used the parametric approach due to its simplicity and accuracy, because the Lagrange multiplier approach gives an off-curve truncation error for each node point on the station curve. The local off -curve truncation error measures the normal displacement of the discrete grid node from the given curve. It is known that the parametric formulation of the grid generation process guarantees that the off-curve truncation error equals zero at these discrete grid nodes. Therefore, the concept of parameter space is introduced in the parametric approach to ensure that the grid points lie upon the given curve. Parameter space consists of an intermediate domain $0 \leq r \leq 1$ such that $\mathbf{r} = \mathbf{r}(\xi)$ is a mapping from logical space to parameter space, and $\mathbf{x} = \mathbf{x}(r)$ is the user-specified mapping from parameter space to physical space (the space in which the hydrodynamic calculations are performed). The following transformation rules apply to the composite map from logical to physical space:

$$\mathbf{x}_\xi = \mathbf{x}_r \cdot \mathbf{r}_\xi, \quad \mathbb{J} = \mathfrak{J}\mathbf{r}_\xi \quad \text{and} \quad g_{11} = \mathbf{g}_{11}\mathbf{r}_\xi^2, \tag{21}$$

where $\mathfrak{J}^{\mathrm{T}} = (\mathbf{x}_r, \mathbf{y}_r, \mathbf{z}_r)$ and $\mathbf{g}_{11} = \mathbf{x}_r \cdot \mathbf{x}_r$. In order to obtain an efficient computational scheme, the first term in equation (19) can be expressed in terms of the logical space derivative (ξ). The variational Euler–Lagrange equation which gives the grid generation equation for a curve can then be expressed in terms of ξ in the following form [see Knupp & Steinberg (1994)]:

$$\left(\frac{\mathscr{H}}{\tilde{w}^2}\right)_\xi - \frac{\mathscr{G} r_{\xi\xi} r_\xi \mathscr{B}}{\tilde{w}^2} - r_\xi \left(\frac{\mathscr{G} r_\xi \mathscr{B}}{\tilde{w}^2}\right)_\xi = 0. \tag{22}$$

In conservative form, this equation may be expressed as

$$\left\{\frac{(\mathscr{H} - \mathscr{G} r_\xi^2 \mathscr{B})}{\tilde{w}^2}\right\}_\xi = 0, \tag{23}$$

where

$$\mathscr{G} = \mathfrak{J}^{\mathrm{T}} \cdot \mathbb{J} = [\mathbf{g}_{11}], \quad \mathbf{g}_{11} = \mathbf{x}_r \cdot \mathbf{x}_r, \quad \mathfrak{J}^{\mathrm{T}} = (x_r, y_r, z_r),$$

$$\mathscr{B} = 2\left(\frac{\partial \mathscr{H}}{\partial g_{11}}\right), \quad g_{11} = \mathbf{g}_{11} \cdot r_\xi^2, \quad \mathbb{J} = (x_\xi, y_\xi, z_\xi);$$

$\tilde{w}$ is a physical weight function, and $\mathscr{H}$ an arbitrary positive real homogeneous function that provides invariance of the grid generator. Letting $\mathscr{H} = g_{11}$, the conservative form of the variational Euler-Lagrange equation becomes

$$\left\{\frac{\mathbf{g}_{11}\, r_{\xi}^{2}}{\tilde{w}^{2}}\right\}_{\xi} = 0. \tag{24}$$

The effect of the physical weight function $\tilde{w}(r)$ (or physical-space weight function $w(\mathbf{x}(\mathbf{r}))$), here, is to reduce (or enlarge) the point spacing x_{ξ} where $\tilde{w}(\mathbf{r})$ is large (or small), and therefore the weight function should be set as some measure of the solution variation or the solution error. One choice can be the curvature of the solution curve. The node points, then, will be concentrated in regions of high curvature of the solution curve (which is desirable in most cases) and near the extremes with a tendency toward equal spacing in regions of zero curvature.

5. HYPERBOLIC TANGENT DISTRIBUTION ON STATION CURVES

In order to optimize the distribution for a given number of grid points, another solution is the boundary-fitted coordinate transformation that clusters points in regions where the dependent variables undergo rapid variation. In our case, these regions result from the body geometry (i.e., very large curvatures or steep corners). In the regions of rapid variation, the clustering can be obtained from either automatic grid generators which solve an elliptic boundary value problem, or one-dimensional stretching functions derived purely algebraically. We define stretching function here as a transformation involving stretching or clustering. In this study, we will use two-sided stretching functions on which the slopes at the two ends of the one-dimensional interval are specified. An interior stretching function is defined as

$$U(\xi) = 0{\cdot}5 + \frac{\tan\left[\Delta z\left(\xi^{*} - \frac{1}{2}\right)\right]}{2\tan\left(\Delta z/2\right)}, \tag{25}$$

where

$$\Delta z = a + \mathrm{i}b, \quad \xi^{*} = \frac{(\xi - \xi_{\min})}{(\xi_{\max} - \xi_{\min})}.$$

The curve parameter ξ changes between $\xi_{\min} \leq \xi \leq \xi_{\max}$, and $0 \leq \xi^{*} \leq 1$. Here $\xi_{\min}$ and $\xi_{\max}$ correspond to keel and deck ends on the station curve, respectively. Assume that S_0 and S_1 are nondimensional slopes at both ends of the space curve. Then B is defined as the nondimensional variable $B = \sqrt{S_0 S_1} = [\sin \Delta z]/\Delta z$. The continuous behaviour of the nondimensional slopes S_0 and S_1, which vary from zero to infinity, is an important criterion in the development of a two-sided stretching function. This is especially necessary to obtain smooth grids constructed algebraically. The method dictates that if $B < 1$, one uses a tangent stretching function with real argument where $\Delta z = a$; and if $B > 1$, one uses a hyperbolic tangent stretching function with pure imaginary argument where $\Delta z = \mathrm{i}b$. Thus, $U(\xi)$ is found once we know the value of a or b using the relation $B = [\sin \Delta z]/\Delta z$. Finally, the resulting grid distribution is determined by using the equation

$$S(\xi) = \frac{S_{\max}\, U(\xi)}{A + (1 - A)U(\xi)}, \tag{26}$$

where $S_{\max}$ = max. length of the curve and $A = \sqrt{S_0/S_1}$.

6. NUMERICAL IMPLEMENTATION

The numerical implementation for an advancing ship starting from rest will be briefly analyzed in the first sub-section. Then, computational details of grid generation on station curves will be introduced in the following sub-section.

6.1. Desingularization

The Indirect Desingularized Boundary Integral Method (Beck *et al.* 1994) uses sources distributed outside the fluid domain, so that the source points never coincide with the collocation or node points, and the integrals are nonsingular. Due to the desingularization we can use simple isolated sources, rather than a distribution over the panel, with the equivalent accuracy. The isolated sources are distributed a small distance above each of the nodes. The distance to the isolated sources is in general given by $L_d = l_d(D_m)^\beta$, where D_m is proportional to the local mesh size and l_d and β are the constant nondimensional parameters to be selected. A detailed study of the performance of DBIEM with respect to the desingularization parameters is reported in Cao *et al.* (1991).

The unknown velocity potential ϕ must be found by solving the mixed boundary value problem at each time. The potential at any point in the fluid domain is given by the distribution of Rankine sources:

$$\phi(\mathbf{x}) = \iint_\Omega \sigma(\mathbf{x}_s)\left(\frac{1}{|\mathbf{x}-\mathbf{x}_s|}\right)\mathrm{d}\Omega, \tag{27}$$

where Ω is the integration surface outside the fluid domain and σ is the source strength to be determined. Applying the relevant boundary conditions with the given potential on the free surface and the known normal velocity on the body, the desingularized indirect boundary integral equations that must be solved at each time-step to determine the unknown source strengths are

$$\iint_\Omega \sigma(\mathbf{x}_s)\left(\frac{1}{|\mathbf{x}_c-\mathbf{x}_s|}\right)\mathrm{d}\Omega = \phi_0(\mathbf{x}_c), \quad (\mathbf{x}_c \in \Gamma_d) \tag{28}$$

and

$$\iint_\Omega \sigma(\mathbf{x}_s)\frac{\partial}{\partial \mathbf{n}}\left(\frac{1}{|\mathbf{x}_c-\mathbf{x}_s|}\right)\mathrm{d}\Omega = \chi(\mathbf{x}_c), \quad (\mathbf{x}_c \in \Gamma_n), \tag{29}$$

where $\mathbf{x}_s$ is a source point on the integration surface, $\mathbf{x}_c$ a field point on the real boundary, ϕ_0 the given potential value at $\mathbf{x}_c$, Γ_d a surface on which ϕ_0 is given, χ the given normal velocity at $\mathbf{x}_c$ and Γ_n a surface on which χ is given.

The integration domain includes the free surface and body surfaces. The desingularized forms of the integral equations (28) and (29) are satisfied at the nodes on the free surface and body surface, leading to a system of N equations for the N unknown source strengths ($\sigma(\mathbf{x}_s)$). Here N represents the total number of unknowns ($N = N_\mathrm{F} + N_B$, where N_F is the number of node points on free surface and N_B is the number of node points on the body). Subsequently, ϕ can be evaluated by equation (27) after solving the resulting equations (28) and (29) simultaneously. Its spatial derivative $\nabla\phi$ can be obtained from the derivative of equation (29), where the differentiation of the Rankine source can be done analytically. In the case of a flat sea bottom, image sources can be used to eliminate the integration over the bottom surface. The hydrodynamic forces and moments acting on the body were computed by integrating the pressure over the instantaneous wetted hull surface,

$$\mathbf{F} = \iint_{S_B} P\mathbf{n}_B \,\mathrm{d}S \tag{30}$$

and

$$\mathbf{M} = \iint_{S_B} P(\mathbf{r} \times \mathbf{n}_B)\,\mathrm{d}S, \tag{31}$$

where

$$\mathbf{n}_B = (n_1, n_2, n_3), \mathbf{r} \times \mathbf{n}_B = (n_4, n_5, n_6);$$

r and $\mathbf{n}_B$ are the position vector and unit normal vector (out of fluid) of the body surface, respectively. Bernoulli's equation gives the total pressure on the wetted body surface

$$\frac{P}{\rho} = -\,gz - \frac{1}{2}\nabla\phi\cdot\nabla\phi - \frac{\delta\phi}{\delta t} + \mathbf{v}\cdot\nabla\phi, \tag{32}$$

where $\delta\phi/\delta t$ is the time derivative of the velocity potential following a moving node on the body and **v** is the velocity of the node relative to the Oxyz system. Earlier studies (Tanizawa 1995; Beck *et al*., 1994) show that a numerical differentiation (e.g., backward difference) for $\delta\phi/\delta t$ can lead to poor estimates of the derivative and possible instabilities. Therefore, in this paper a new boundary value problem (BVP) was set up for $\delta\phi/\delta t$ using the given $\delta\phi/\delta t$ on the free surface (Bernoulli's equation) and $\partial(\delta\phi/\delta t)\partial n = 0$ on the body surface. The $\delta\phi/\delta t$ on the free surface was calculated from the dynamic free-surface condition given by equations (12) and (14). The time derivative of the velocity potential was then obtained by solving the second matrix equation at each time-step with the same influence coefficient matrix. This scheme can also be directly applied to the freely floating body simulation.

6.1.1. *Iterative solver and preconditioning*

The resulting influence coefficient matrix is obtained from equations (28) and (29) and it is large, dense and unsymmetrical. In solving such a large matrix, the iterative methods generally have advantage over the Gauss-elimination-type matrix solvers. The iterative matrix solver called the Generalized Minimal Residual Method (Saad & Schultz 1986) is here used for solving the system at each time-step. GMRES is an extension of MINRES (which is only applicable to symmetric systems) to unsymmetrical systems, and generates a sequence of orthogonal vectors keeping the previously computed vectors in the orthogonal sequence. The most popular form of GMRES is based on the modified Gram–Schmidt procedure, and uses restarts to control storage requirements. Another alternative for a partitioned block- matrix system is to use the block Jacobi method. It is known that the block iterative method is faster than the point method on most serial computers. In this study, the Jacobi iterative algorithm developed for blocked matrices is used alternatively:

$$A^{(i,i)}\ X^{(i)}_{(k+1)} = -\sum_{(j\neq i)} A^{(i,j)} X^{(j)}_{(k)} + B^{(i)}, \tag{33}$$

where X and B are partitioned commensurately with A. Thus, carrying out one iteration requires the solution of the previous systems of equation (33) with coefficient matrices $A^{(i,j)}$.

Assuming that $X_{(k)}^{(j)}$ and $B^{(i)}$ contain initially guessed values and the boundary values, respectively, the unknowns $X_{(k)}^{(j)}$ on each row are updated simultaneously using the values from the previous iteration. This is called the *line Jacobi method*. Since each $A^{(i,i)}$ is assumed to be nonsingular, a parallel implementation of the above method is done using $L_i U_i$ decomposition of $A^{(i,i)}$.

A preconditioner is a matrix that accelerates the convergence rate of an iterative method. We know that the convergence rate depends on the spectral properties of the coefficient matrix. Therefore, the main purpose of the preconditioner is to transform the linear system into one that is equivalent in the sense that it has the same solution, but it has more favourable spectral properties. In our computations, we used a preconditioning matrix $\hat{A}$ such that $\hat{A}^{-1}A$ closely approximates the identity matrix, yielding a strongly diagonally dominant matrix and small condition number. In general, it is not easy to find an optimum $\hat{A}$ matrix for a dense, nonsymmetric A matrix. In this study, however, we used $\hat{A}$ as A^o, i.e., A at $t = 0$. Even if the free-surface boundary and hence A change with time, A^o is still quite a good approximation to A, especially when the boundary integral is desingularized. Another alternative to obtain more favourable spectral properties of the coefficient matrix is to minimize the condition number using one-sided scaling based on the Hölder p-norm. This technique is applied to the current problem as an alternative to the GMRES method. A brief summary of this technique is as follows. Let $A \in \mathbb{R}^{m \times n}$ have full rank and let $D_k \subset \mathbb{R}^{k \times k}$ denote the set of nonsingular diagonal matrices, and define

$$D_c := diag(\|A(:,j)\|_p)^{-1}, \quad D_R := diag(\|A(i,:)\|_p)^{-1}. \tag{34}$$

Then

$$k_p(AD_c) \leq n^{1-1/p} \min_{D \in D_n} k_p(AD) \tag{35}$$

and

$$k_p(D_R A) \leq m^{1/p} \min_{D \in D_m} k_p(DA). \tag{36}$$

For $p = \infty$, inequality (36) confirms that row equilibration is an optimal row scaling strategy; similarly, for $p = 1$, column equilibration is the best column scaling, by inequality (35). Known as the *Van der Sluis* theorem, this shows that row and column equilibration produce condition numbers within factors $\sqrt{m}$ and $\sqrt{n}$, respectively, of the minimum 2-norm condition numbers achievable by row and column scaling. The above procedure was tested by applying the scaling on the partitioned matrices of A rather than the A matrix itself. Test results show that the reduction of the condition number for each partitioned submatrices is around $1/(MN)^{1/4}$, where $(M \times N)$ is the size of the submatrices. This reduction in the condition number gives better convergence in the computations. As a result, scaling of the influence matrix can also be used to reduce the iteration number, especially in the block Jacobi method, and hence to accelerate the convergence speed.

6.1.2. *Saw-tooth instability and regridding*

The so-called *sawtooth* instability was first introduced by Longuet-Higgins and Cokelet (1976) during their two-dimensional breaking wave simulations. They suggested that a possible cause of the instability was partly numerical and partly physical. They, therefore, employed a smoothing technique to suppress the development of the sawtooth instability by very short waves. Later researchers pointed out that a possible cause of the instability was the concentration of Lagrangian points in the region of higher gradients, when the minimum grid size cannot be effectively controlled for a given time-step. For this reason, a similar regridding algorithm developed by Dommermuth and Yue (1987) is here used to eliminate such instabilities. Thus, a new set of uniformly spaced Lagrangian points is created on the free surface and body surface after every time step. The boundary values on the new set of nodal points are then redistributed by using a cubic-spline interpolation technique. Then, the fourth–fifth order Runge–Kutta–Fehlberg time-integration scheme is employed to continue the time stepping. Using this algorithm, most of the instabilities are completely removed and no artificial smoothing is required.

The main disadvantage of regridding is the potential loss of resolution which is usually provided by more closely spaced Lagrangian points in the areas of large gradients. The advantages of regridding over artificial smoothing, however, particularly in the present context, are substantial, in that the smoothing cannot be straightforwardly applied at the body-free-surface intersection line and the crossing of Lagrangian points can be more easily controlled.

6.2. Numerical Grid Generation

Suppose that the body node locations x_i on the station curve need to be computed, $0 \leq i \leq M$, where M is a given positive integer and $x_0 = \mathbf{a}$ and $x_M = \mathbf{b}$ are the first and last points on the station curve. The finite difference approximation to the differential equation (24) results in an algebraic equation called *the stencil equation* [see Knupp & Steinberg (1994)],

$$r_{i-1}\frac{L_i}{(\Delta\xi)^2} + r_i\frac{C_i}{(\Delta\xi)^2} + r_{i+1}\frac{R_i}{(\Delta\xi)^2} = 0 \tag{37}$$

where

$$L_i = \Gamma_{i-1/2}, \quad R_i = \Gamma_{i+1/2}, \quad C_i = -(L_i + R_i),$$

$$\Delta\xi = \tfrac{1}{M}, \qquad \xi_i = i\Delta\xi,\ 0 \leq i \leq M.$$

The lagged coefficients are evaluated as follows:

$$\Gamma_{i+1/2} = \left[\frac{\mathbf{g}_{11}}{\tilde{w}^2}\right]_{i+1/2}(r_{i+1} - r_i). \tag{38}$$

The first factor on the right is merely a function of the parameter r, so it may be evaluated at the half-point by

$$r_{i+1/2} = \frac{(r_{i+1} + r_i)}{2}. \tag{39}$$

As usual, $i = 1, \ldots, M-1$ are the interior points and $r_0 = 0$, $r_M = 1$ are the boundary conditions. Assume, initially, r_i is equal to x_i. Any problem where the L_i, R_i, or r_i depend on the grid, that is, on the x_i, is nonlinear. The problem is solved using a *nonlinear iteration* procedure. A small tolerance parameter ε that determines the accuracy of the solution must be given. To start the iteration, an initial grid x_i^{old} must be generated, say using linear interpolation. Then, a nonlinear iteration loop starts with evaluating L_i, C_i, R_i, and $\mathbf{g}_{ii}$ using x_i^{old}. The next step is to solve the resulting linear system for $x_i = x_i^{\mathrm{new}}$ using a tridiagonal solver. Finally, one computes $\delta = \max_i |x_i^{\text{new}} - x_i^{\text{old}}|$. If $\delta > \varepsilon$ then one sets $x_i^{\text{old}} = x_i^{\text{new}}$ and loops, otherwise one quits the loop.

In order to apply the numerical curve grid generation techniques discussed above, we need shape information for each station along the hull which includes coordinate transformation, curvature distribution, and girth length. A ship hull form is generated such that a body form defined by an offset is fitted by a single B-spline surface using defining vertex points. The hull form is generated on the computer using offsets, and verified by the curvature porcupines which provide second-order (C^2) continuity on the station curves. Special hull definition routines are devised for accessing shape information for any arbitrary station along the hull. This is performed using a Newton root finding search technique.

7. NUMERICAL VERIFICATION

For the present transient nonlinear ship wave computations, the numerical error can be controlled by the number of node points in the solution domain, the mesh size, the desingularization distance (l_d), the error tolerances for the Runge–Kutta–Fehlberg integration scheme and the iterative matrix solver. Extensive numerical tests proved that, as long as the matrix error tolerance is less than 10^{-7} on 64-bit machines (such as CRAY J90, C90, or SGI Power Challenge), the Runge–Kutta–Fehlberg error tolerance less than 10^{-3}, and if the mesh has at least 12 nodes per wavelength, the boundary value problem can be solved with high accuracy. The time-step used for the hydrodynamic computations is directly related to integration scheme and its order. Therefore, $\Delta t = 0{\cdot}1$ s is used for the computations. To check the convergence with the number of nodes, three different discretizations on the free surface, $N_F = 1246$, 2420, and 3840, were tested (Kim, Celebi & Kim 1998). The convergence rate was measured by

$$E_{\text{rms}} = \sqrt{\frac{1}{N_T}\sum_{i=1}^{N_T}(\sigma_M^i - \sigma_{M-1}^i)^2},$$

where M denotes the number of iterations, N_T is the total number of nodes and σ^i denotes the unknown source strength at the ith node point. For $M = 12$, $E_{\text{rms}} = 10^{-3}$, 10^{-6}, and 10^{-8}, respectively for $N_F = 1246$, 2420, and 3840, which shows that more accurate results can be obtained with finer discretization (see Figure 2). Numerical tests show that the distance of the isolated sources tends to get smaller value for finer discretization and if the value of L_d is too large or too small for the same discretization, then the number of iterations for convergence increases significantly, or the solution diverges during the iterative solution of the influence matrix system.

Typical ship sections were tested using both grid-generation techniques. The basic difference between the two curve grid generators is that the Vinokur distribution can manipulate the grids simply by choosing the stretching parameters at two ends; however variational grid generation distributes the grids in terms of the slope gradient of the space curve. Initial computations show that the distribution of the nodes using the hyperbolic tangent method depends on choosing the stretching parameters for each station curve along the ship hull. In the case of 100 stations, choosing and testing of the 200 stretching parameters would be an additional burden besides the node distribution itself. Figure 3 shows the two-sided hyperbolic tangent distribution and the variational grid distribution on the station curve, for the same section function. Stretching parameters $\xi_{\text{deck}} = 0{\cdot}5$ and $\xi_{\text{keel}} = 0{\cdot}05$ are used in the hyperbolic tangent distribution to get the similar node placement. Figure 4 shows the comparison of the two techniques on a typical ship section using a given number of node points on each station. The results show that both methods

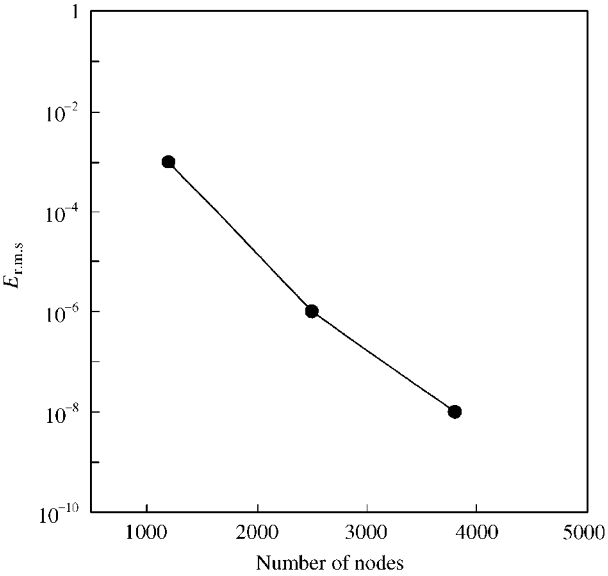

Figure 2. Convergence rate with respect to node number ($L_d = 0{\cdot}65$): —●—, R.m.s. error.

distribute nodes similarly, with some extra effort needed in the hyperbolic distribution (such as choosing stretching parameters). Using the curvature as the space weight function appears to be a good choice. As can be seen in Figure 4, the node points are properly distributed. On the zero curvature regions (calculated in parametric space), there is an equal chord length spacing. On the other hand, nodes tend to accumulate in regions of high curvature. For the stations tested, the weight function formulation seems to work effectively. More testing for a larger array of section types is needed to verify that using the curvature as a space weight function is an optimal node distributor.

8. NONLINEAR SHIP-WAVE COMPUTATIONS

Figure 5 shows the coordinate system used for the computations. The Wigley hull form was used in order to verify the mathematical procedure of surface fitting and node distribution, since the answers could be compared with previous computations (Beck *et al.* 1994). The chosen hull has a length-to-beam ratio of 10, a beam-to-draft ratio of 1·6 and the following equation for its hull surface:

$$y = \pm \frac{B}{2}\left\{1 - \left(\frac{2x}{L}\right)^2\right\}\left\{1 - \left(\frac{z}{H}\right)^2\right\}, \tag{40}$$

where L is the length, B/2 the half-beam, and H the draft.

Two wall-sided Wigley hulls with different bow and stern profiles were used to compute the wave resistance and ship-generated free-surface waves [Figure 6(a, b)]. The second Wigley hull form with angled bow and stern profiles has been created from the standard Wigley hull form by changing the bow and stern profiles, corresponding waterlines, and stations. This was accomplished by creating a Wigley hull using a CAD program and then the bow and stern profiles were pulled below the calm waterline and stretched above it. The reason for using a B-spline representation for the standard Wigley hull form is to test the

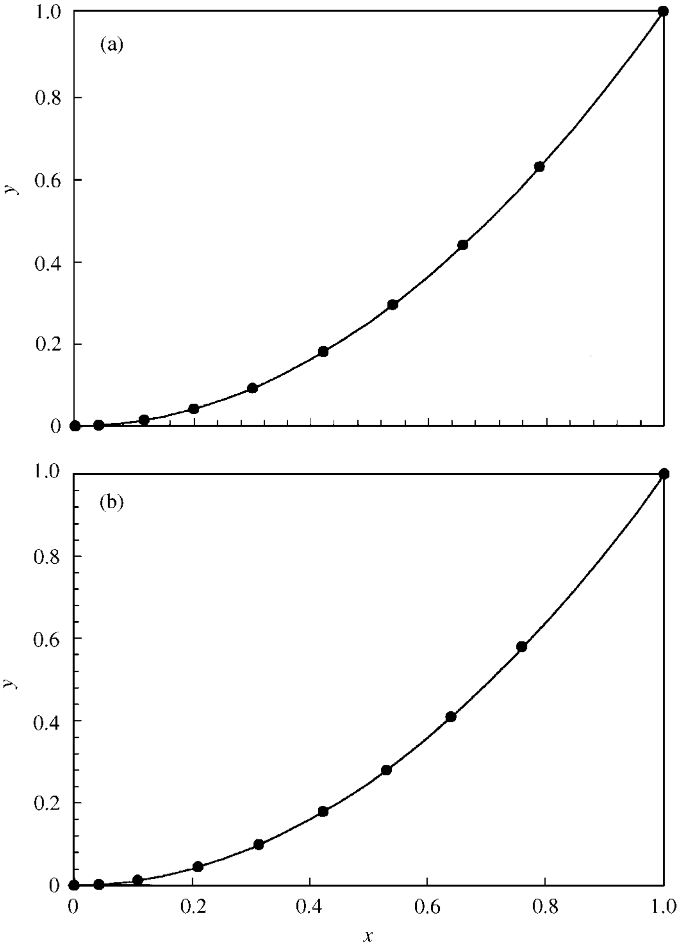

Figure 3. (a) Hyperbolic tangent: ——, $y = x^2$; ●, $\xi_{\mathrm{deck}} = 0.5$, $\xi_{\mathrm{keel}} = 0{\cdot}05$. (b) Variational grid distribution on section function $y = x^2$: ——, $y = x^2$; ●, weight function $w(x) = [w(x)]^{1/2}$.

node distribution techniques. We may then compare the B-spline results with the previous analytic results of Beck *et al.* (1994) and verify the accuracy of the B-spline surface representation and node distributions. Both models are started from rest with their velocity at any time given by

$$U_0(t) = U_0(1 - \mathrm{e}^{-0{\cdot}05t^2}). \tag{41}$$

The parameter 0·05 was found to be suitable for the range of model scales investigated here. The reason for a Gaussian start-up was that it has zero acceleration at $t = 0$, but another start-up formulation such as hyperbolic tangent could be chosen.

The normalized wave profiles along the hull at $t\sqrt{g/L} = 20$ (where $t = 28{\cdot}5\,\mathrm{s}$, corresponding to steady-state conditions being reached) are given in Figures 7(a) and 7(b) for two Froude numbers; (the Froude number is defined by $\mathrm{Fn} = V/\sqrt{gL}$ where V, L, and g are the velocity, the characteristic length of the body, and the gravitational acceleration, respectively). The experimental result for a standard 2·5 m Wigley hull fixed in sinkage and trim are taken from Noblesse & McCarthy (1983). A Wigley hull with an angled bow and stern and

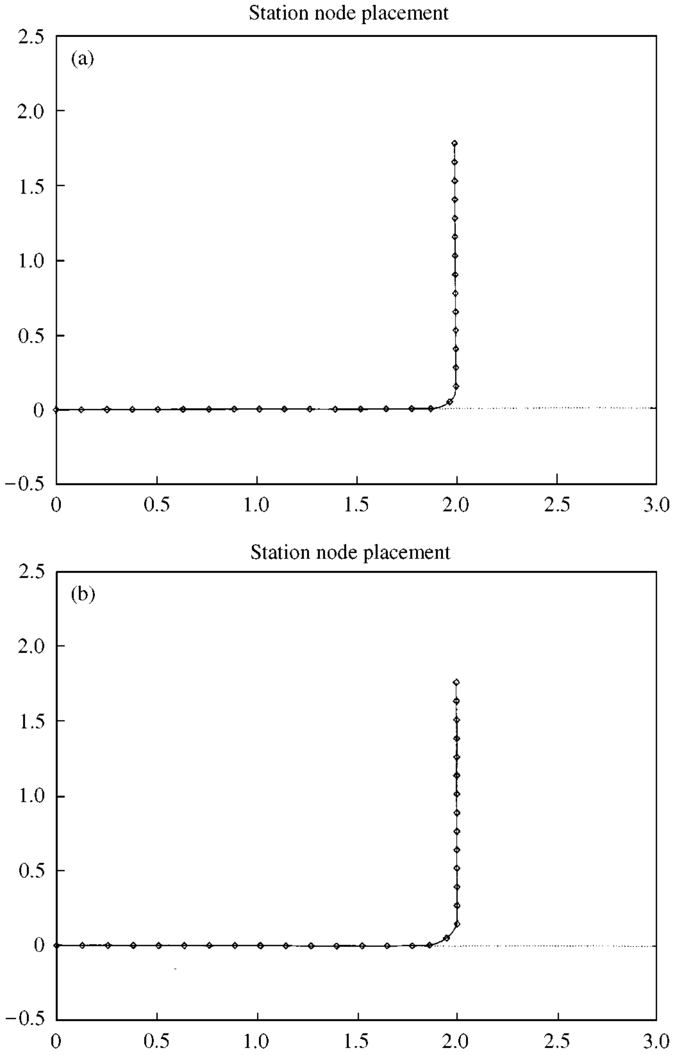

Figure 4. (a) Hyperbolic tangent: ——, ship-hull section; ◇, $\xi_{\text{deck}} = 0{\cdot}1$, $\xi_{\text{keel}} = 0{\cdot}15$. (b) Variational grid distribution on a typical ship-hull section: ——, ship-hull section; ◇, weight function $w(x) = [w(x)]^{1\cdot 7}$.

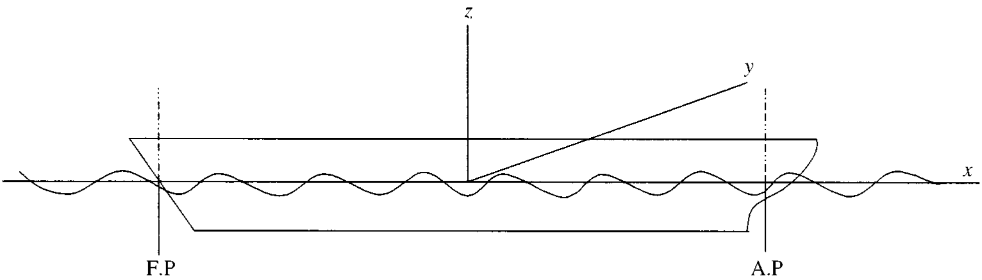

Figure 5. Coordinate system.

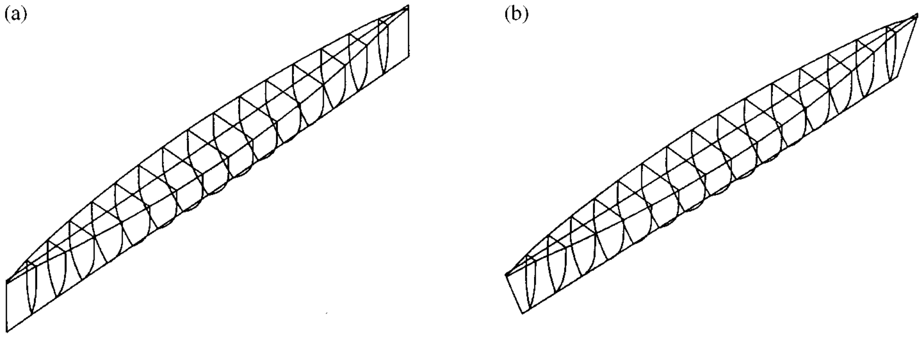

Figure 6. (a) Standard wall-sided Wigley hull (b) Wigley hull with sloped bow and stern.

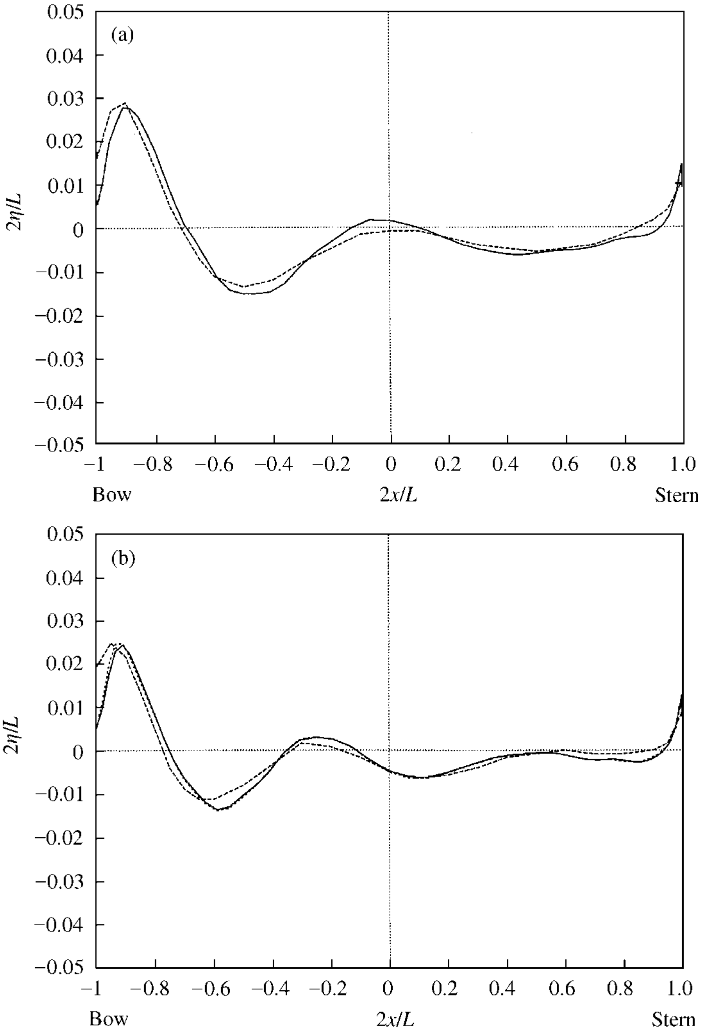

Figure 7. Comparison of the wave profiles along the Wigley hull, $t\sqrt{g/L} = 20$. (a) Fn = 0·289: ——, present method, - - -, experiment. (b) Fn = 0·25: ——, present method; - - - experiment; · · · ·, Beck *et al.* (1994).

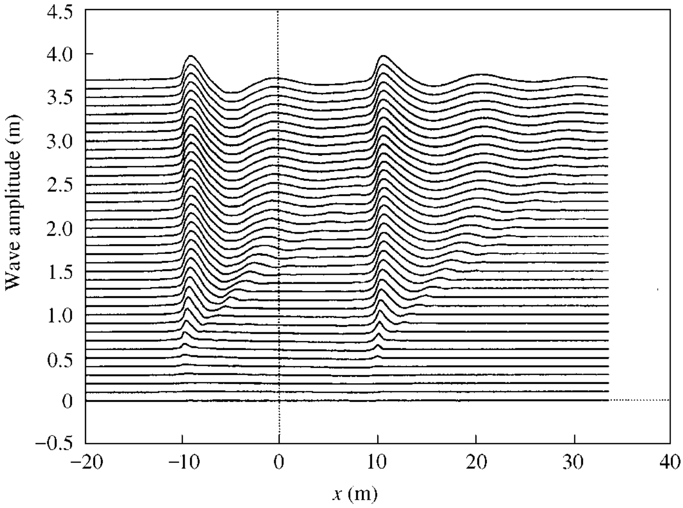

Figure 8. Development of the wave system along the Wigley hull surface, Fn = 0·289. The wave profiles are at equally spaced time-steps in the interval $t\sqrt{g/L} = (0,\ 13{\cdot}5)$.

the standard Wigley hull are used in Figures 7(a) and 7(b), respectively. The approximate value of the total change in wetted volume due to the raked bow and stern is around 5%. There is generally good agreement except for a slight difference and shifting in the bow wave amplitude. There is also a slight overestimate in the mid-ship region, and low estimate in the front ($2x/L = -0{\cdot}6$ to $-0{\cdot}4$) and stern ($2x/L = 0{\cdot}8$ to $0{\cdot}9$) regions. Comparisons and test results showed that adaptive node distribution and B-spline approximation of the Wigley hull are quite accurate and acceptable for estimating the wave formation around the hull. The development with time of the wave system along the plane of symmetry and around the hull (angled bow and stern) for Fn = 0·289 is given in Figure 8. It can be observed that the wave amplitude increases with time and approaches an almost steady form. The wave pattern generated by a Wigley hull with angled bow and stern is investigated in Figure 9 for Fn = 0·289 at time $t\sqrt{g/L} = 17$ (where $t = 24{\cdot}3$ s). The results confirm that the bow and stern are two basic sources of wave formation. There is a front disturbance field at the bow, and transverse wave formation starting from the bow region due to the forward body displacement.

Finally, as an illustration of the effectiveness of the adaptive algorithm, the development with time of the total wave resistance of a Wigley hull with an angled bow and stern is shown in Figure 10 for Fn = 0·289. The components of total resistance force are nondimensionalized by $\frac{1}{2}\rho U_0^2 S_0$, where U_0 is the final velocity and S_0 is the nominal wetted surface ($S_0 = 0{\cdot}148L^2$ in this case). The biggest contribution to the total seems to come from the $-\rho U_0(t)\partial\phi/\partial x$ component. The average value of total wave resistance for Fn = 0·289 approaches a value 5·9% higher than the experimental result (marked EXP in Figure 10) given by Noblesse & McCarthy (1983). One possible reason for overestimating the total wave resistance in Figure 10 is the lack of viscous effects in the numerical model. At low speeds, the waves made by the ship are small, and the resistance is predominantly viscous in character. Note that, for a non-viscous and incompressible fluid, the ship will experience only the wave-making resistance. As a result of this, the net fore-and-aft force resulting from the pressure distribution over the wetted hull surface gives the wave-making resistance. However, the shear stress in the fluid domain is assumed zero, thus the frictional resistance

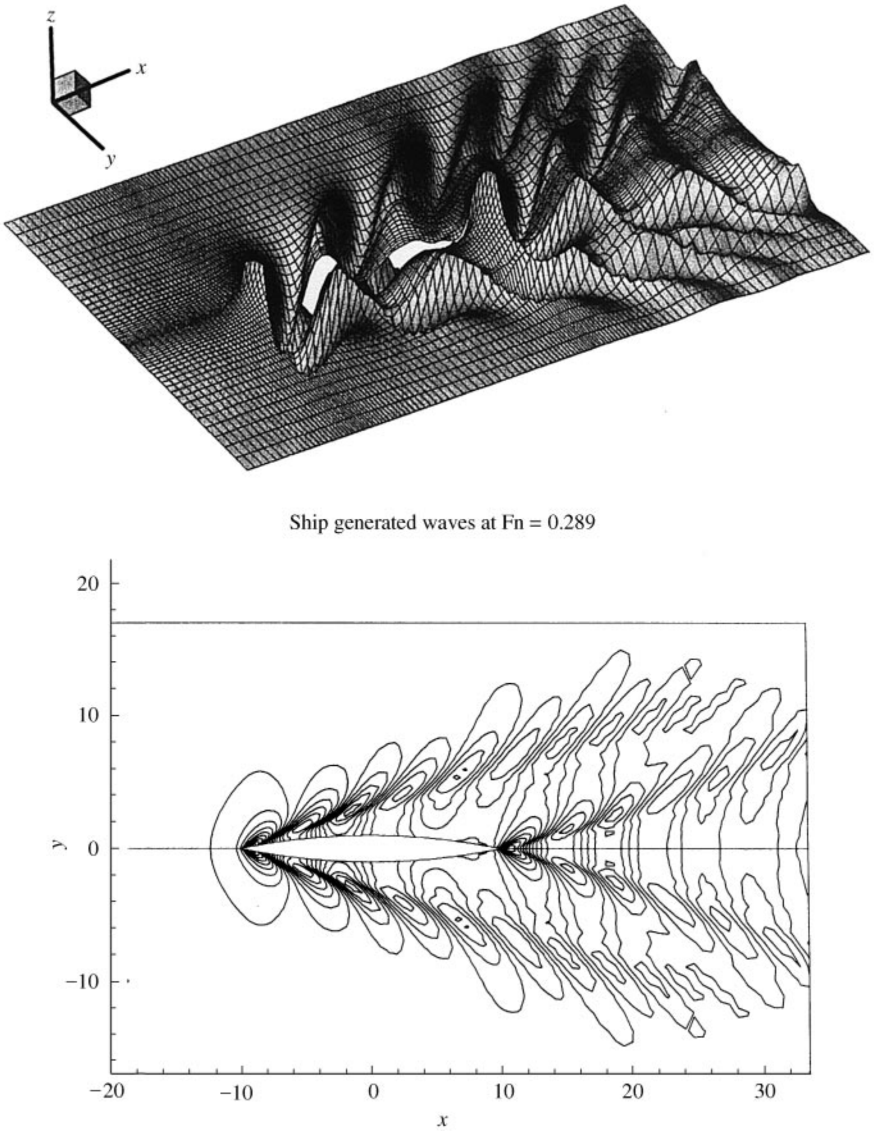

Figure 9. Free-surface waves for Wigley hull with angled bow and stern, $N_F = 3267$, $N_B = 535$, $t\sqrt{g/L} = 17$.

is neglected. Another possible reason is the scale effect on the wave-making resistance during the experiment. This is directly related to neglecting the effects of viscosity and depends on the Reynolds number and therefore on the size of the model. The model behaves as though it were longer than its actual length, and this is undoubtedly due mostly to the virtual lengthening of the form due to the viscous boundary layer. The experimental results may contain some additional errors due to surface tension and blockage effects. Surface tension causes capillary waves and other effects and blockage effects increase the frictional resistance because the walls and the bottom of the basin make the water move faster past the ship.

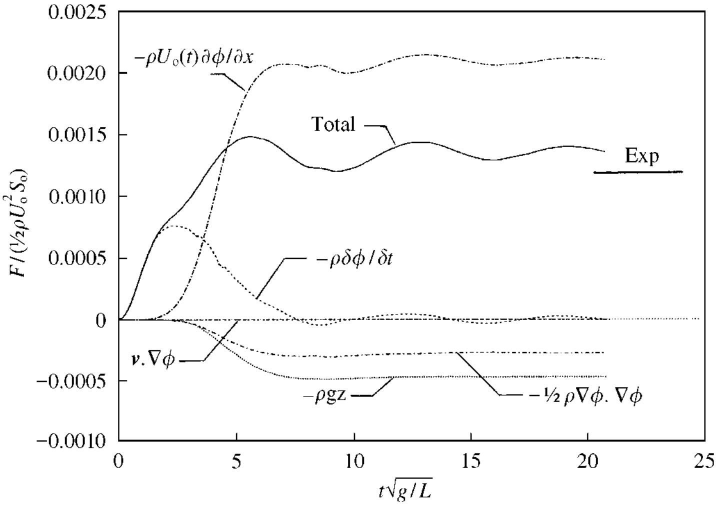

Figure 10. Wave resistance components for Wigley hull, Fn = 0·289

Most of the CPU time (80%) is consumed in regridding the free surface and body, and reconstruction and solution of the influence matrix system. In order to reduce the CPU time, local dynamic station allocations instead of the global allocation need to be used. Furthermore, improving the performance of the matrix construction and solution is vital to speed up the computation. In this regard, more efficient iterative solvers with effective preconditioners (such as one-sided diagonal scaling or modified incomplete factorization) are being developed. Alternatively, a more efficient multipole acceleration scheme such as presented by Scorpio *et al.* (1996) can be employed to further speed up the solution procedure.

9. CONCLUDING REMARKS

An indirect desingularized boundary integral method is used for the solution of the transient nonlinear ship-wave problem in three dimensions. Two adaptive grid-generation techniques are proposed and compared. Both techniques redistribute the node points automatically on the instantaneous wetted hull surface at each major integration time-step. Two types of hull form are used: (a) a Wigley hull, and (b) a Wigley hull with raked bow and stern. The second hull form is created in a parametric space using B-spline surface approximations. Node points are distributed either by a weight function or by a two-sided hyperbolic tangent technique. The method allows the computations to be carried out on arbitrary hull forms. In this study, the hull form is represented with a single B-spline hull surface. Thus it is difficult, in general, to add bulbous bows and transom sterns with the required accuracy. Therefore, for more complex hull forms, the next step will be to increase the B-spline hull surface representation to more than one surface. The locations of the source points near the bow and stern regions directly affect the convergence of the iteration procedure. Two different iterative matrix solvers have been developed and used here for fast convergence and accurate computations of the influence matrix system. Specially designed preconditioners and matrix scaling have been developed and employed to improve the convergence speed.

ACKNOWLEDGMENT

Computations were partially performed at Texas A & M and San Diego supercomputer centers.

References

Beck, R. F. 1994 Time domain computations for floating bodies. *Journal of Applied Ocean Research*, **16,** 267–282.

Beck, R. F., Cao, Y. & Lee, T. H. 1993 Fully nonlinear water wave computations using the desingularized method, *Proceedings of 6th International Conference on Numerical Ship Hydrodynamics*, The University of Iowa.

Beck, R. F., Cao, Y., Scorpio, S. & Schultz, W. 1994 Nonlinear ship motion computations using the desingularized method, *20th Symposium on Naval Hydrodynamics*, Santa Barbara, CA, U.S.A.

Brackbill, J. & Saltzman, J. 1982 Adaptive zoning for singular problems in 2-D. *Journal of Computational Physics*, **46,** 342–368.

Cao, Y., Schultz, W. W. & Beck, R. F. 1991 Three dimensional desingularized boundary integral methods for potential problems. *International Journal of Numerical Methods in Fluids* **12,** 785–803.

Cao, Y., Lee, T. & Beck, R. F. 1992 Computation of nonlinear waves generated by floating bodies. *7th International Workshop on Water Waves and Floating Bodies*, pp. 47–52, Val de Reuil, France.

Castillo, J. 1987 On variational grid generation. Ph.D. dissertation, University of New Mexico, Albuquerque, NM, U.S.A.

Celebi, M. S. 1997 *Computation of Transient Fully Nonlinear Free Surface Waves and Wave–body Interactions*. Ph.D. Dissertation, Texas A&M University, College Station, Texas, U.S.A.

Celebi, M. S. & Beck R. F. 1997 Geometric modelling for fully nonlinear ship-wave interactions. *Journal of Ship Research* **41,** 10–18.

Celebi, M. S. & Kim, M. H. 1997 Nonlinear wave body interactions in a numerical wave tank. *12th International Workshop on Water Waves and Floating Bodies*, France.

Celebi, M. S., Kim, M. H. & Beck R. F. 1998 Fully nonlinear three-dimensional numerical wave tank simulation. *Journal of Ship Research*, **42,** 33–45.

Dommermuth, D. G. & Yue, D. K. P. 1987 Numerical simulations of non linear axisymmetric flows with a free surface. *Journal of Fluid Mechanics* **178,** 195–219.

Kim, M. H., Celebi, M. S. & Kim, D. J. 1998 Fully nonlinear interactions of waves with a three-dimensional body in uniform currents. *Applied Ocean Research* **20,** 309–321.

Knupp, P. & Steinberg, S. 1994 *Fundamentals of Grid Generation*. Boca Raton, FL: CRC Press.

Longuet-Higgins, M. S. & Cokelet, C. D. 1976 The deformation of steep surface waves on water: I. A numerical method of computation. *Proceedings of Royal Society*, London, A**350,** 1–26.

Noblesse, F. & McCarthy, J. H. 1983 *Proceedings of the Second DTNSRDC Workshop on Ship Wave-Resistance Computations*, pp. 5–35. DTNSRDC, Maryland U.S.A.

Saad, Y. & Schultz, M. H. 1986 GMRES: A Generalized minimal residual algorithm for solving non-symmetric linear systems. *SIAM Journal of Scientific and Statistical Computing* **7,** 856–869.

Saltzman, J. 1986 Variational methods for generating meshes on surfaces in 3-D. *Journal of Computational Physics*, **63,** 1–19.

Scorpio, S. M., Beck, R. F. & Korsmeyer, F. T. 1996 Nonlinear water wave computations using a multipole accelerated, desingularized method. *Proceedings of 21st International Symposium on Naval Hydrodynamics*, Tronheim, Norway.

Steinberg, S. & Roache, P. 1991 Anomalies in grid generation. AIAA paper 88–3740.

Steinbrenner, J. P., Chawner, J. R. & Fouts, C. L. 1990 Multiple block grid generation in the interactive environment. 90–1602. AIAA paper.

Tanizawa, K 1995 A nonlinear simulation method of three-dimensional body motions in waves. *Journal of SNAJ*, **178,** 179–191.

Vinokur, M. 1980 On one-dimensional stretching functions for finite-difference calculations. NASA CR 3313.